

WEST Search History

DATE: Wednesday, February 02, 2005

Hide?	<u>Set Name</u>	<u>Query</u>	<u>Hit Count</u>
	<i>DB=PGPB,USPT,USOC; PLUR=YES; OP=ADJ</i>		
<input type="checkbox"/>	L12	L11 and vertex	10
<input type="checkbox"/>	L11	L10 and trac\$	134
<input type="checkbox"/>	L10	L1 and L2	243
<input type="checkbox"/>	L9	L2 and (reach\$ near edge)	18
<input type="checkbox"/>	L8	L2 and vertex	29
	<i>DB=EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=ADJ</i>		
<input type="checkbox"/>	L7	L4 and (reach\$ near edge)	0
<input type="checkbox"/>	L6	L4 and ring	1
<input type="checkbox"/>	L5	L4 and reachability	1
<input type="checkbox"/>	L4	L3 and (edge or vertex)	21
<input type="checkbox"/>	L3	L2	121
	<i>DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=ADJ</i>		
<input type="checkbox"/>	L2	(control-flow or control flow or data-flow or dataflow data flow) near graph	622
	<i>DB=PGPB,USPT,USOC; PLUR=YES; OP=ADJ</i>		
<input type="checkbox"/>	L1	717/124-133,136-144,154-161.ccls.	3707

END OF SEARCH HISTORY



Web Images Groups^{New!} News Froogle more »

"data flow" "control flow" edge vertex reachabi

Search

Advanced Search
Preferences

Web Results 1 - 10 of about 246 for "**data flow**" "**control flow**" **edge vertex** **reachability**. (0.37 seconds)

Citations: A Formal Model of Program Dependences and Its ...

... A flow dependence **edge** runs from a **vertex** that represents a ... of the program, are in turn divided into **data flow** and **control flow** criteria, although the ...

citeseer.ist.psu.edu/context/29073/0 - 29k - Jan 31, 2005 - [Cached](#) - [Similar pages](#)

Citations: Flow graph reducibility - Hecht, Ullman (ResearchIndex)

... Tar74] 2.2.2 Local analysis Local **data flow** analysis determines ... Let $G = N; E; s$ be a **control flow** graph and ... $2N$. The T1 transformation removes the **edge** $(u; u \dots$
citeseer.ist.psu.edu/context/143465/0 - 20k - [Cached](#) - [Similar pages](#)

[[More results from citeseer.ist.psu.edu](#)]

Undecidability of context-sensitive data-independence analysis

... variable, (2)an **edge** from **vertex** ... and flow-sensitive interprocedural **data flow** analysis, Proceedings ... Linear-time subtransitive **control flow** analysis, Proceedings ...

portal.acm.org/citation.cfm?id=345137 - [Similar pages](#)

[PDF] On the Limit of **Control Flow** Analysis for Regression Test ...

File Format: PDF/Adobe Acrobat - [View as HTML](#)

... In P's **control flow** graph G, each **vertex** represents a basic block of instructions and each **edge** represents a control transition between blocks. ...

research.microsoft.com/~tball/papers/issta98.pdf - [Similar pages](#)

[PDF] Undecidability of Context-Sensitive Data-Dependence Analysis

File Format: PDF/Adobe Acrobat - [View as HTML](#)

... data dependence, set-based analysis, set constraints, **control-flow** analysis ... but a generalization in which the **edge** labels are ... 25,32]: A path P from **vertex** s to ...

www.cs.wisc.edu/wpis/papers/toplas00.pdf - [Similar pages](#)

[PDF] Reusable Idioms and Patterns in Graph Transformation Languages

File Format: PDF/Adobe Acrobat - [View as HTML](#)

... The hierarchical **data flow** network and flat representation will ... branching construct to choose between **control flow** paths ... is referred as a negative pattern **edge**. ...

www.softmetaware.com/oopsla2004/agrawal.pdf - [Similar pages](#)

Control flow graph information

... to several compiler optimization#**Data-flow** optimizations, compiler ... of the program, and represents all alternatives of **control flow**. ... caused by a back **edge** to a ...

www.searchspaniel.com/index.php/Control_flow_graph - 27k - [Cached](#) - [Similar pages](#)

[PDF] Reusable Idioms and Patterns in Graph Transformation Languages

File Format: PDF/Adobe Acrobat - [View as HTML](#)

... The hierarchical **data flow** network and flat representation will serves ... object: a **vertex** (or an **edge**) conforms to ... construct to choose between **control flow** paths ...

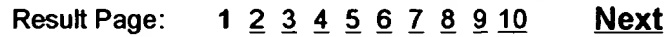
www.isis.vanderbilt.edu/publications/archive/Agrawal_A_9_28_2004_Reusable_I.pdf - [Similar pages](#)

[PDF] Dynamic Program Slicing

File Format: PDF/Adobe Acrobat - [View as HTML](#)

... trol flow, a control-dependence **edge** @om **vertex** u; to **vertex** u_j implies that V\$ is the nearest inverse dominator of v_i in the **control flow** paph of the program ...

[PDF] [System-Dependence-Graph-Based Slicing of Programs with Arbitrary ...](#)
File Format: PDF/Adobe Acrobat - [View as HTML](#)
... PNRCs, ConstructSDG (line 1b) performs **data-flow** analysis (eg ... labeled with the target **vertex** of that **edge**. ... P. Finally, an ACFG contains **control-flow** edges that ...
www.cc.gatech.edu/aristotle/Publications/Papers/icse99.pdf - [Similar pages](#)



 **Free! Google Desktop Search:** Search your own computer.

"data flow" "control flow" edge vertex

[Google Home](#) - [Advertising Programs](#) - [Business Solutions](#) - [About Google](#)

<http://www.google.com/search?hl=en&lr=&q=%22data+flow%22+%22control+flow%22...> 2/2/05

[Web](#)
[Images](#)
[Video](#) NEW!
[Directory](#)
[Local](#)
[News](#)
[Products](#)

YAHOO! SEARCH

[Shortcuts](#)
[Advanced Search](#)
[Preferences](#)

Search Results

Results **1 - 10** of about **20** for **data flow control flow edge vertex reachable** - 0.20 se

<http://www-rocq.inria.fr/~pop/orc-gen.html>

wn has no vertex VINDEIX16 Get_Verx() Get the first vertex ... cxx implement a control flow graph for edge f
analysis" to find out which ...

www-rocq.inria.fr/~pop/orc-gen.html - 257k - [Cached](#) - [More from this site](#)

<http://www.math.utah.edu/pub/tex/bib/intjparallelprogram.bib>

control driven execution; **control flow** primitives; data dependencies ... on static **data flow** computers", journal :
aths", treatment ...

www.math.utah.edu/pub/tex/bib/intjparallelprogram.bib - 416k - [Cached](#) - [More from this site](#)

[ibTeX bibliography intjparallelprogram.bib](#)

control driven execution; **control flow** primitives; data dependencies ... on static **data flow** computers", journal :
aths", treatment ...

www.math.utah.edu/ftp/pub/tex/bib/intjparallelprogram.html - 509k - [Cached](#) - [More from this site](#)

[dependence Graphs and Program Slicing](#)

is a dependence **edge** from each declaration **vertex** to each of ... even though it is **reachable** by traversing edg
raphs (CFGs), and symbol ...

www.grammatech.com/research/slicing/slicingWhitepaper.html - 41k - [Cached](#) - [More from this site](#)

[odeSurfer Technology Overview \(PDF\)](#)

There is a dependence **edge** from each declaration **vertex** to each of ... even though it is **reachable** by traversi
STs), **control flow** graphs (CFGs), ...

www.grammatech.com/research/slicing/slicingWhitePaper.pdf - 404k - [View as html](#) - [More from this site](#)

[ll Classes \(FLEX API documentation\)](#)

ClassHierarchy doesn't contain a **reachable** method which overrides it ... for building simple **data flow** analyses
odes in the **control-flow** graph ...

msser-magoo.lcs.mit.edu/Harpoon/doc/allclasses-noframe.html - 278k - [Cached](#) - [More from this site](#)

<http://www.cs.utexas.edu/ftp/pub/AI-Lab/tech-reports/ABSTRACTS>

197-259 Tilt Aftereffects in a Self-Organizing Model of the Primary Visual Cortex James A. ... is generated from th
rd algebraic manipulation of equations associated ...

www.cs.utexas.edu/ftp/pub/AI-Lab/tech-reports/ABSTRACTS - 264k - [Cached](#) - [More from this site](#)

[itations: The Design and Analysis of Computer Algorithms - Aho, Hopcroft, Ullman \(eBizSearch\)](#)

. Aho, J. Hopcroft, J. Ullman, The Design and Analysis of Computer Algorithms. Addison-Wesley (1974). ... objec
ese nodes ... A sample program. **Data flow edge**: If data propagate from ... of Table 1. **Control Flow Declaration**

anther.smeal.psu.edu/context/14720/0 - 64k - [Cached](#) - [More from this site](#)

<http://elib.cs.sfu.ca/Collections/CMPT/MajorBibs/beebe-bib/intjparallelprogram.bib>

control driven execution; **control flow** primitives; data dependencies ... on static **data flow** computers", journal :
aths", treatment ...

elib.cs.sfu.ca/Collections/CMPT/MajorBibs/beebe-bib/... - 349k - [Cached](#) - [More from this site](#)

[omputer BE Syllabus](#)

layer, Framing error **control**, **Flow** control and Elementary data ... structural, Behavioral, and **data flow** modelir

documents, **reachable** from it ...

www.aitpune.com/computerbesyllabus.asp - 110k - [Cached](#) - [More from this site](#)

Results Page:

1 2 ► **Next**

Web | **Images** | **Video** NEW! | **Directory** | **Local** | **News** | **Products**

Your Search:

Search

Help us improve your search experience. [Send us feedback](#).

Copyright © 2005 Yahoo! Inc. All rights reserved. [Privacy Policy](#) - [Terms of Service](#) - [Submit Your Site](#) - [Job Openings](#)

Searching for **control flow and trace and vertex**.

Restrict to: [Header](#) [Title](#) Order by: [Expected citations](#) [Hubs](#) [Usage](#) [Date](#) Try: [Google \(CiteSeer\)](#)
[Google \(Web\)](#) [Yahoo!](#) [MSN](#) [CSB](#) [DBLP](#)

22 documents found. Order: number of citations.

[A Survey of Program Slicing Techniques - Tip \(1995\) \(Correct\) \(230 citations\)](#)

some specific test case. Procedures, arbitrary **control flow**, composite datatypes and pointers, and relevant statements, according to data flow and **control flow** dependences. Only statically available on the values computed at statement t , the user may **trace** back from the **vertex** corresponding to statement plaslab.cis.nctu.edu.tw/eriol/TR/Tip94.ps.Z

One or more of the query terms is very common - only partial results have been returned. Try [Google \(CiteSeer\)](#).

[Optimally Profiling and Tracing Programs - Ball, Larus \(1994\) \(Correct\) \(111 citations\)](#)

frequency of each block or edge in a program's **control-flow** graph. We have implemented the algorithms in a for inserting monitoring code to profile and **trace** programs. These algorithms greatly reduce the www.bell-labs.com/~tball/papers/optprotrace.ps.gz

[Branch Prediction For Free - Ball, Larus \(1993\) \(Correct\) \(84 citations\)](#)

many programs with complex conditional **control flow**. Our measurements show that a perfect static methods for measuring this quantity with **trace**-based methods and show why **trace**-based methods for each procedure in the executable file. Each **vertex** in the **control flow** graph represents a basic ftp.cs.wisc.edu/tech-reports/reports/93/tr1137.ps.Z

[Slicing Concurrent Programs: A Graph-Theoretical Approach - Cheng \(1993\) \(Correct\) \(25 citations\)](#)

because a concurrent program has multiple **control flows**, multiple data flows, and interprocess by solving data flow equations over an execution **trace** of the program [17]Agrawal and Horgan a concurrent program can be simply reduced to a **vertex** reachability problem in its PDN representation. www.aise.ics.saitama-u.ac.jp/~cheng/pub/Slicing.AADEBUG93.ps

[Virtual Reality and Parallel Systems Performance Analysis - Reed \(1995\) \(Correct\) \(23 citations\)](#)

the same code, albeit with data-dependent **control flow**. For example, in most ray tracing codes the and the duration of the read)A typical event **trace** implicitly defines 5-10 dynamic performance grid-based data (i.e. one or more metrics at each **vertex** of an n -dimensional grid)In consequence, the vibes.cs.uiuc.edu/Publications/Papers/ComputerVR.ps.gz

[From Symptom to Cause: Localizing Errors in Counterexample.. - Ball, Naik, Rajamani \(2003\) \(Correct\) \(17 citations\)](#)

a note on future work. 2. PRELIMINARIES 2.1 **Control Flow** Graphs We represent programs abstractly via to Cause: Localizing Errors in Counterexample Traces Thomas Ball Microsoft Research xenon.stanford.edu/~mhn/pubs/popl03.pdf

[Visualizing Program Slices - Ball, Eick \(1994\) \(Correct\) \(17 citations\)](#)

statement since it executes regardless of the **control-flow** path through the procedure. Combined, the data computed by examining the execution's address **trace** (the tradeoff is that not all dependences in the backwards closure from 'return(y)'includes every **vertex** in the graph except for ' $x = y$ 'and ' $p[x]$ ' www.bell-labs.com/user/tball/papers/vl94.ps.gz

[Interprocedural Path Profiling - Melski, Reps \(1998\) \(Correct\) \(9 citations\)](#)

Algorithms, Measurement Additional Key Words: **Control-flow** graph, instruction tracing, instrumentation, drive partial inlining, either for interprocedural **trace** scheduling or for an interprocedural extension of of the edge labels along each path from P 's entry **vertex** to P 's exit **vertex** falls in the range $[0:\text{number}]$ www.cs.wisc.edu/wpis/papers/tr1382.ps

[Software Verification with BLAST - Thomas Henzinger Ranjit \(2003\) \(Correct\) \(7 citations\)](#)

Internally, C programs are represented as **control flow** automata (CFA)which are **control flow** graphs

is feasible, Blast outputs the path as an error **trace**, otherwise, it uses the infeasibility of the the program. Each node of the tree is labeled by a **vertex** of the CFA and a formula, called the reachable
www-cad.eecs.berkeley.edu/~tah/Publications/software_verification_with_blast.ps

Dynamic Slicing of Object-Oriented Programs - Zhao (1998) (Correct) (6 citations)

program slices through static data flow and **control flow** analysis and is valid for all possible
computes program slices through static data flow and **control flow** analysis and is valid for all possible
However, by carefully examining the execution **trace** of the program with input argv[1]3, we can
www.fit.ac.jp/~zhao/personal/ps/se-tr98-119.ps.gz

List Scheduling in the Presence of Branches: A Theoretical.. - Franco Gasperoni And (1998) (Correct) (3 citations)

by looking at branching task systems whose **control flow** graph is acyclic. First, we define an
an innovative compilation technique called **trace** scheduling, that went beyond the conditional jump
an acyclic single entry, single exit di-graph with **vertex** set O [fig such that no operation in G has
www-ds.e-technik.uni-dortmund.de/WEB-D/Forschung/reports/tr0596.ps

Timestamping Messages in Synchronous Computations - Vijay Garg And (2002) (Correct) (1 citation)

communication requires buffer management and **flow control** mechanisms. Implementation of synchronous
such as POET [13]XPVM [12]and Object-Level **Trace** [11]It is also used in the area of global
of size less than or equal to the size of the **vertex** cover of the communication topology of the
www.ece.utexas.edu/~garg/dist/dcs02.ps

The Offset Cube: An Optoelectronic Interconnection Network - Scott Wills (1994) (Correct) (1 citation)

parameters and assumptions: 1. Wormhole **flow control** is used. 2. Eight virtual channels are
performance analysis with random traffic and **trace**-driven workloads, comparison with k-ary 3-cubes)
where k is the length of each side of a square **vertex** array on each layer and L is the number of
exodo.upr.clu.edu/~jcruz/Papers/PCRCW.ps

Graphics Hardware (2002), pp. 1–10 - Thomas Ertl Wolfgang (Correct)

and near-future GPU's due to the lack of **control flow** in the fragment program, resulting in a large
efficient structures whose processing requires **flow control**)A simple ray tracing system we built using
renderer augmented with the ray engine could **trace** the rays necessary to achieve effects currently
graphics.cs.uiuc.edu/~nacarr/papers/RayEngine-gh02.pdf

Results from a Large-Scale Study of Performance Optimization.. - Binkley, Harman (2003) (Correct)

and most widely used internal graphs is the **control-flow** graph developed for use within a compiler.
work [16]A large variety of external graphs **trace** their original to object orientation and, in
With the exception of call statements, a single **vertex** represents predicates (e.g.from if and while
www.brunel.ac.uk/~csstmmh2/scam03.ps

Cache Line Coloring Procedure Placement Using Real and.. - Hashemi, Kaeli, Calder (Correct)

discuss call graph construction with the aid of **control flow** graphs. We also survey past work in program
our algorithm and demonstrate its merit through **trace**-driven cache simulation. In Section 2 we discuss
Each procedure will be mapped to a single **vertex**, and each call path between two procedures will
www.ece.neu.edu/info/architecture/publications/toplas.ps.gz

Cache Line Coloring Using Real and Estimated Profiles - Hashemi, Kalamatianos.. (Correct)

discuss call graph construction with the aid of **control flow** graphs. We also survey past work in program
our algorithm and demonstrate its merit through **trace**-driven cache simulation. In Section 2 we discuss
Each procedure will be mapped to a single **vertex**, and each call path between two procedures will
www.ece.neu.edu/info/architecture/publications/dec2.ps.gz

GURU: A retargetable cfg-based program reorganizer - Ramadan, Gupta (1995) (Correct)

section we present background information on **control flow** graphs and program profiling. We also survey
taken branches. 2.3 Tools for Profiling and **Trace** Generation A variety of tools are available for
flow graph (CFG) is a directed graph, where each **vertex** in the graph represents a basic block of
ftp.cs.clemson.edu/techreports/95-112.ps.Z

First 20 documents [Next 20](#)



Welcome
United States Patent and Trademark Office


[Help](#) [FAQ](#) [Terms](#) [IEEE Peer Review](#)
[Quick Links](#)

Welcome to IEEE Xplore®

- ☐ Home
- ☐ What Can I Access?
- ☐ Log-out

Tables of Contents

- ☐ Journals & Magazines
- ☐ Conference Proceedings
- ☐ Standards

Search

- ☐ By Author
- ☐ Basic
- ☐ Advanced
- ☐ CrossRef

Member Services

- ☐ Join IEEE
- ☐ Establish IEEE Web Account
- ☐ Access the IEEE Member Digital Library

IEEE Enterprise

- ☐ Access the IEEE Enterprise File Cabinet

Your search matched **2** of **1123491** documents.

A maximum of **500** results are displayed, **15** to a page, sorted by **Relevance Descending** order.

Refine This Search:

You may refine your search by editing the current search expression or entering new one in the text box.

☐ Check to search within this result set

Results Key:

JNL = Journal or Magazine **CNF** = Conference **STD** = Standard

1 Concurrent self-checking for microprogrammed control units: an analytical survey

Demidenko, S.N.; Levin, E.M.; Lever, K.V.;

Computers and Digital Techniques, IEE Proceedings E [see also Computers and Digital Techniques, IEE Proceedings-], Volume: 138, Issue: 6, Nov. 1991
Pages:377 - 388

[\[Abstract\]](#) [\[PDF Full-Text \(928 KB\)\]](#) **IEE JNL**

2 Concrete aircraft routing algorithms

Loubieres, P.; Chemla, J.;

Digital Avionics Systems, 2001. DASC. The 20th Conference, Volume: 2, 14-Oct. 2001
Pages:7F3/1 - 7F3/10 vol.2

[\[Abstract\]](#) [\[PDF Full-Text \(627 KB\)\]](#) **IEEE CNF**

 Print Format

[Home](#) | [Log-out](#) | [Journals](#) | [Conference Proceedings](#) | [Standards](#) | [Search by Author](#) | [Basic Search](#) | [Advanced Search](#) | [Join IEEE](#) | [Web Account](#) | [New this week](#) | [OPAC Linking Information](#) | [Your Feedback](#) | [Technical Support](#) | [Email Alerting](#) | [No Robots Please](#) | [Release Notes](#) | [IEEE Online Publications](#) | [Help](#) | [FAQ](#) | [Terms](#) | [Back to Top](#)

Copyright © 2004 IEEE — All rights reserved



US Patent & Trademark Office

[Subscribe](#) (Full Service) [Register](#) (Limited Service, Free) [Login](#)

Search: ☒ The ACM Digital Library ☐ The Guide

+ "control flow" +graph +tracing +vertex +ring



[Feedback](#) [Report a problem](#) [Satisfaction survey](#)

Terms used control flow graph tracing vertex ring

Found 10 of 150,138

Sort results
by

relevance



Display
results

expanded form



[Save results to a Binder](#)

[Search Tips](#)

☐ Open results in a new
window

Try an [Advanced Search](#)

Try this search in [The ACM Guide](#)

Results 1 - 10 of 10

Relevance scale ☐ ☐ ☐ ☐ ☐

1 [On the limit of control flow analysis for regression test selection](#)

Thomas Ball

March 1998 **ACM SIGSOFT Software Engineering Notes , Proceedings of the 1998 ACM SIGSOFT international symposium on Software testing and analysis**, Volume 23 Issue 2

Full text available: pdf(838.83 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Automated analyses for regression test selection (RTS) attempt to determine if a modified program, when run on a test t , will have the same behavior as an old version of the program run on t , but without running the new program on t . RTS analyses must confront a price/performance tradeoff: a more precise analysis might be able to eliminate more tests, but could take much longer to run. We focus on the application of control flow analysis and control flow coverage, relatively ...

Keywords: control flow analysis, coverage, profiling, regression testing

2 [A Survey of Some Theoretical Aspects of Multiprocessing](#)

J. L. Baer

January 1973 **ACM Computing Surveys (CSUR)**, Volume 5 Issue 1

Full text available: pdf(4.05 MB)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

3 [Automatic array alignment in data-parallel programs](#)

Siddhartha Chatterjee, John R. Gilbert, Robert Schreiber, Shang-Hua Teng

March 1993 **Proceedings of the 20th ACM SIGPLAN-SIGACT symposium on Principles of programming languages**

Full text available: pdf(1.34 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Data-parallel languages like Fortran 90 express parallelism in the form of operations on data aggregates such as arrays. Misalignment of the operands of an array operation can reduce program performance on a distributed-memory parallel machine by requiring nonlocal data accesses. Determining array alignments that reduce communication is therefore a key issue in compiling such languages. We present a framework for the automatic determination of array alignments in data-parallel la ...

- 4 Detecting stable properties of networks in concurrent logic programming languages
Vijay A. Saraswat, Kenneth Kahn, David Weinbaum
January 1988 **Proceedings of the seventh annual ACM Symposium on Principles of distributed computing**

Full text available:  [pdf\(1.72 MB\)](#)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

- 5 Session 2: dynamic program analysis: Isolating cause-effect chains from computer programs

Andreas Zeller

November 2002 **ACM SIGSOFT Software Engineering Notes**, Volume 27 Issue 6

Full text available:  [pdf\(1.12 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Consider the execution of a failing program as a sequence of program states. Each state induces the following state, up to the failure. Which variables and values of a program state are relevant for the failure? We show how the *Delta Debugging* algorithm isolates the relevant variables and values by systematically narrowing the state difference between a passing run and a failing run--by assessing the outcome of altered executions to determine whether a change in the program state makes a difference ...

Keywords: automated debugging, program comprehension, testing, tracing

- 6 Data path debugging: data-oriented debugging for a concurrent programming language

Wenwey Hseush, Gail E. Kaiser

November 1988 **ACM SIGPLAN Notices , Proceedings of the 1988 ACM SIGPLAN and SIGOPS workshop on Parallel and distributed debugging**, Volume 24 Issue 1

Full text available:  [pdf\(1.12 MB\)](#)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

- 7 Safety consideration for storage allocation optimizations

D. R. Chase

June 1988 **ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 1988 conference on Programming Language design and Implementation**, Volume 23 Issue 7

Full text available:  [pdf\(1.06 MB\)](#)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

- 8 Beyond induction variables

Michael Wolfe

July 1992 **ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 1992 conference on Programming language design and implementation**, Volume 27 Issue 7

Full text available:  [pdf\(1.20 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Induction variable detection is usually closely tied to the strength reduction optimization. This paper studies induction variable analysis from a different perspective, that of finding induction variables for data dependence analysis. While classical induction variable analysis techniques have been used successfully up to now, we have found a simple algorithm based on the Static Single Assignment form of a program that finds all linear induction variables in a loop. Moreover, this algorithm ...

- 9 Sparse matrix solvers on the GPU: conjugate gradients and multigrid

Jeff Bolz, Ian Farmer, Eitan Grinspun, Peter Schröder

Full text available:  [pdf\(753.59 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Many computer graphics applications require high-intensity numerical simulation. We show that such computations can be performed efficiently on the GPU, which we regard as a full function *streaming* processor with high floating-point performance. We implemented two basic, broadly useful, computational kernels: a *sparse matrix conjugate gradient solver* and a regular-grid *multigrid solver*. Real time applications ranging from mesh smoothing and parameterization to fluid solvers ...

Keywords: GPU computing, Navier-Stokes, conjugate gradient, fluid simulation, mesh smoothing, multigrid, numerical simulation

10 An extensible probe architecture for network protocol performance measurement

G. Robert Malan, Farnam Jahanian

October 1998 **ACM SIGCOMM Computer Communication Review , Proceedings of the ACM SIGCOMM '98 conference on Applications, technologies, architectures, and protocols for computer communication**, Volume 28 Issue 4

Full text available:  [pdf\(1.83 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

This paper describes the architecture and implementation of Windmill, a passive network protocol performance measurement tool. Windmill enables experimenters to measure a broad range of protocol performance metrics by both reconstructing application-level network protocols and exposing the underlying protocol layers' events. Windmill is split into three functional components: a dynamically compiled Windmill Protocol Filter (WPF), a set of abstract protocol modules, and an extensible experiment e ...

Keywords: online analysis, packet filter, passive measurement, protocol performance

Results 1 - 10 of 10

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2005 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads:  [Adobe Acrobat](#)  [QuickTime](#)  [Windows Media Player](#)  [Real Player](#)